https://doi.org/10.31449/inf.v46i7.4280 Informatica **46** (2022) 131-144 **131**

A Multi-label Classification of Disaster-Related Tweets with Enhanced Word Embedding Ensemble Convolutional Neural Network Model

E. Arathi[1]*, S.Sasikala[1]
E-mail: aarthi.devpal@gmail.com, sasikalarams@gmail.com
1 Computer Science, IDE, University of Madras Chennai, Tamil Nadu, India.
* Corresponding author

Keywords: Twitter data classification, Social Media, Convolutional Neural Network (CNN), Ensemble Deep Learning, Embeddings from Language Models (ELMo), Recurrent Neural Network (RNN), multi-label classification.

Received: August 8, 2022

Abstract:

Recently, adopting machine learning techniques to automate the identification and classification of event-related tweets has been beneficial in times of crisis. Word embeddings are the most effective word vectors for NLP processing using deep learning classifiers. This research proposes a novel method with the Enhanced Embedding from Language Model (EnELMo) for classifying tweets as different categories with higher classification accuracy and precision for the rapid rescue action in the disaster scenario. The proposed Enhanced Word Embedding Ensemble Convolutional Neural Network Model(EWECNN) method comprises an Enhanced ELMo module with crisis lexicon to create Crisis word vectors, a Novel ELMo-CNN Architecture module for feature extraction (ECA), and an effective multi-label classification of text using Crisis Word Vector specific CNN-RNN(CWV-CRNN) stacks. These functional modules are intended to improve the classification. Among the various approaches discussed, the proposed method outperforms the classification of microblog texts with an accuracy of 93.46 percent and the F1-Score of 92.99 percent for multi-classification of tweets, which is higher than other methods discussed in this study. The proposed multi-label classification of disaster-related text facilitates faster rescue action in a crisis scenario.

Povzetek: Predstavljena je nova metoda za iskanje uspešnega reševanja v primeru katastrof na osnovi analiz socialnih omrežij z ansamblom nevronskih mrež (EWECNN).

1 Introduction

Disasters have an impact on societies all over the world because they can strike without warning and cause massive damage. The reaction and recovery of communities during all disaster phases rely on their level of preparedness [1]. During a crisis, those affected, authorities, and volunteers seek actionable information to aid in damage restoration and rescue operations. Timely and accurate information is critical for humanitarian and government efforts to save lives and gain access to those affected [2]. The phenomenal utilization and additional accomplishments make the social media platform widely used in crisis dissemination.

Recent advances in Artificial Intelligence have resulted in the development of machine learning techniques to aid in automating Twitter data multi-classification. Text classification is the process of automatically assigning tags to predefined classes based on the content of texts. Different types of tweets are posted during a disaster, such as rumors and spam, emotional information, and other types of data that need to be classified by NLP. In conventional methods, different features are hand-crafted to classify tweets of a specific type in times of crisis. Various deep learning neural network algorithms, such as CNN and RNN, achieve high accuracy in crisis-related multi-classification challenges [4][5][6]. However, more research and inventions are required to improve social media text data classifications. Data collection, preprocessing, feature selection, Construction of a new classification methodology, Training, hyper-parameter fine-tuning, and evaluation of the proposed method are the customary phases of a text classification system [7]. The microblog messages are categorized under eight categories: Disaster kind, Location, Dead and Injured People, Help Request, Infrastructure damage, Search and Rescue, Weather-related information, and Non-relevant information to reach victims of disasters more quickly and efficiently. Using a context-specific Language Model and domain-specific ensemble CRNN, a new multi-label text classifier is developed in this work. Extensive research is conducted for the multi-label categorization of tweets with standard CrisisNLP datasets of different disaster events, evaluated using performance metrics. This study compares various multi-classification approaches with the proposed method.

This paper is structured as follows: The second section covers relevant studies on classifying Twitter messages for various goals. The third section presents an enELMo embedding model employing the ensemble CNN-RNN stack classifier. The experiment is described in section four. The fifth section elaborates on the model's results and analysis of its performance. Finally, Section 6 concludes the paper.

2 Related Works

2.1 Crisis-related Tweets Classification

Recent studies have shown the increasing importance of social media during emergencies and how broadcasting information via social media can improve situational awareness during a crisis. These authors [8,9,10] were the first to look into and analyze the use of microblogs and information lifecycles during crisis scenarios. They studied the behavior of microbloggers by conducting a qualitative analysis of tweets published during a flooding incident. The authors of [11] employed the bags-of-words method to locate a crisis's data. Various grammatical elements, like Parts-Of-Speech (POS), are influenced by the vocabulary used in Twitter posts. In the proposed work, an Artificial Intelligence Disaster Response (AIDR) system was developed using uni-gram and bi-gram features to classify crisis-related information during a disaster. The authors of [12] used the informative terms from tweets as characteristics to recognize resource tweets during a disaster. Based on low-level features of lexical and syntactic characteristics, models for disaster situational awareness were developed in [13,14].

The process of automatically extracting information from tweets has brought much attention to word embedding. The work proposed in [15, 16] used CNN with word embedding and demonstrated superior performance to hand-crafted features in disaster-related tasks. In the study [17,18], CNN and MLP-CNN with word embedding are used to categorize the data linked to crises. The skip-gram model of the word2vec tool was applied [19, 20, 21] to extract information from an extensive corpus consisting of almost 57,908 tweets. During the disaster event, the authors of [22] developed a deep learning model using the word embedding to detect informative tweets related to the catastrophe for speedier actions.

2.2 Multi-Label Classifications of tweets

A set of comparable multi-label text classification methods are taken here to juxtapose the performance of the proposed method. They are, A new big data approach for topic classification and sentiment analysis of Twitter data (BDACSA)[20], A pattern-based approach for multi-class sentiment analysis in Twitter(PAMSA) [23], and improved classification of crisis-related data on Twitter using contextual representations (CRICCD [24], Multi-level aspect-based sentiment classification of Twitter data: using the hybrid approach in deep learning (MASC) [25] and Sentiment classification from multi-class imbalanced Twitter data using binarization (BMSC) [26]. This section confers a limpid analysis of the methodologies used and the advantages and limitations of these existing methods. The Naive Bayes classifier (HL-NBC) proposed by the authors of [27] can be used to classify microblog text into several categories and filter out irrelevant tweets. This experiment tested the Naive Bayes classifier model with Lexicon, unigram, and bigram features. The HL-NBC method improves sentiment classification and achieves an accuracy of 82 percent; however, processing time and cross-lingual categorization are limitations, and the context of the text cannot be found effectively using this approach.

SENTA, a user-friendly tool for classifying text from microblogging websites into seven distinct sentimental categories, was presented in this paper [28]. A customizable feature selection option was designed to extract the features such as Sentiment features, Punctuation features, Syntax/Stylistic Features, and semantic features to preprocess the text for classification. Compared to the text's binary and ternary classification, this model gives a multi-class classification rate of 60.2%, while the binary and ternary classification rates are both 80.1%. To the advantage of this work, there is an abundance of configurable feature classifications. However, this method's average accuracy for multi-class classification is a noted limitation.

The authors of [29] suggested that the intense classifier with ELMo embedding's model achieved a higher metric in terms of performance than the regular Support Vector Machine (SVM) for the multi-classification of disasters-related tweets. To arrive at this conclusion, the researchers used datasets such as those gathered from multiple sources like CrisisNLP, CrisisLex, and AIDR Twitter standard datasets from the earthquakes in Nepal and California and Typhoon Hagupit. An embedding layer can generate word vectors from input text. The dense classifier then accurately predicts the crisis and other tweet texts by processing the word vectors at 82.3 percent. The drawbacks are the necessity of contextual feature selection and better processing time.

Using a new classifier MuLeHyABSC and a feature ranking process, this study [30] claims to perform classification of Twitter data based on multi-level aspect-based classification. MASC employs an Artificial Neural Network Multilayer Perceptron (ANMP) to improve classification performance. Several existing machine learning classifiers are combined with MuLeHyABSC. The evaluation process used the benchmark datasets STC, TAS, FGD, ATC, and STS. the performance of the MASC method secured higher scores in terms of Accuracy, Precision, Recall, and F-Score. Higher classification accuracy and precision are the proven advantages of this method. At the same time, the increased processing time is the limitation of the MASC

method. The higher processing time makes the application MASC lag in the real-time classification processes.
The authors of [31] presented the proposed methodology as one-vs-one binary decomposition and dimension reduction. In addition to Weighted multi-class reconstruction, a stable preprocessing method was introduced to detect minority classes with the MBSC method. SemEval2016 Message Polarity classification dataset was used in the MBSC method. The modules of MBSC were tested with the standard baseline classifiers. The performance metric of MBSC work was measured in terms of Geometric Mean for multi-class classification. Better classification performance is the advantage of this method. Table 1 provides a summary of the related methodologies, their limitations, and the benefits of the existing methods.

Table 1: An outline of the methodologies, advantages, and limitations of the existing methods

Author	**Work**	**Methodology**	**Accuracy %**	**Advantages**	**Limitations**
Anisha. Rodrigues et.al.	A new big data approach for topic classification and sentiment analysis of Twitter data[20]	Hybrid Lexicon-Naïve Bayesian Classifier	82	Processing time	Moderate Accuracy
Mondher Bouaziz et.al.	A pattern-based approach for multi-class sentiment analysis in Twitter .[21]	Tokenization, Lemmatization, and Generating negation vectors	80.1	Highly configurable	Average Accuracy
Sreenivasulu Madichetty et.al	Improved classification of crisis-related data on Twitter using contextual representations[22]	CNN dense classifier with ELMo embedding for feature extraction.	82.3	Better Accuracy	Higher Processing Time
Sadaf Hussain Janjua et.al.	Multi-level aspect-based sentiment classification of Twitter data: using a hybrid approach in deep learning[23]	Artificial Neural Network Multilayer Perceptron	80.38	Higher Accuracy	Higher Processing Time
Bartosz Krawczyk et.al.	Sentiment classification from multi-class imbalanced Twitter data using binarization[24]	one-vs-one binary decomposition and dimension reduction	66.36 (G-mean)	Average metrics for the classification	Higher Processing Time

The above methods discussed are based on different features of the tweet text, such as unigram. Bigram. N-gram features, Sentiment features, Punctuation features, Syntactic/stylistic features, and semantic features are used for the classification. Various feature extraction studies show that word embedding based on language models is more contextual than the other methods. The above studies reveal that the deep learning models outperform the baseline classifiers in terms of accuracy and processing time, as well as the need for a better classifier for effective multi-label classification in the disaster scenario. The proposed model works on the text as the feature using Crisis word vector-specific ELMo embedding with ensemble CNN classifier for the multi-classification of the microblog text, which creates contextualized feature vector and feature map, resulting in improved performance of the classifier.

3 Proposed Methodology

The proposed EWECNN method was devised with three underlying functional modules. Enhanced-ELMo handles disaster-related vectors; Enhanced ELMo-CNN Architecture extracts context-related features, and Crisis Word vector-specific CNN and RNN stacks classify text into multiple classes and significantly improve this multi-label classification. Instead of converting text as word vectors and running the algorithm, this model focuses on creating context-related, disaster word-specific vectors as a whole sentence rather than single words. This section describes the fundamental functional building components in further detail. Figure I displays the whole flowchart for the EWECNN approach. The algorithm I: EWECNN algorithm explains the methodology in a stepwise procedure.

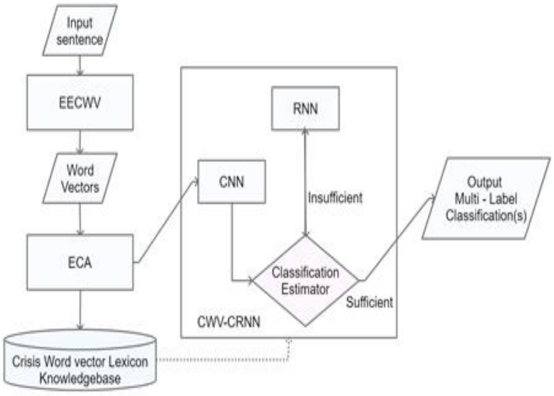

Figure 1: EWECNN Flow Diagram

Algorithm I : EWECNN Algorithm

The proposed EWECNN method for multi-label classification of tweets with crisisNLP dataset
Algorithm I : EWECNN(dataset d)
Begin
// input layer- creating word embeddings using context based language model .
load dataset d
create training, validation, and test sets of data
For each tweet t in d
begin
clean_tweet<-preprocess_tweet(t)
feature_vector <- enELMo(clean_tweet)
end
// enELMo-CNN layer
feature_map = ECA(feature_vector)
// Multi-label classification with ensemble CNN and RNN.
z <- CWV_CRNN(feature_map)
labels<-output(z)
end

3.1 Enhanced ELMo to generate crisis word vectors (EECWV)

This module is the model's input layer, which adds the disaster's context to the word vector for categorization, then transferred to the following phase. For the convolutional net, vectors are generated using embeddings. ELMo can build a contextually rich representation of words and dynamically alter the representation of words, hence resolving polysemous words. A deep bidirectional language model (biLM) is pre-trained on a large crisis lexicon corpus[32] to generate these word vectors. Adding these vectors above each input word for each end task significantly improves the algorithm's performance. enELMo representations are derived from crisis-related word vectors using the RLSTM encoder. The ELMo parses a sequence of word vectors at a time to eliminate the ambiguous meaning which misleads in classification. This context-biased smart purport selection is the primary reason to designate the enELMo model here. The steps are listed as an algorithm II – enELMO algorithm.

Algorithm II: enELMO algorithm
This algorithm process the feature vector of the given dataset using enhanced ELMo
Algorithm enELMo(preprocess_tweet)
Begin
rlstm<-lstm(crisis_word_vector) using equation(4)
crisis_based_embedding<-ELMo(rlstm) using equation(3)
feature_vector<crisis_based_embedding(preprocess _tweet)
return feature_vector
end

ELMo is often suitable for domain-specific categorization. This model achieves domain specificity by interacting with the intermediate layer of biLM models that are initialized by crisis word vectors. For every token t_k, a set of 2L+1 depictions for L-Layer using the following equation.

$$R_k = \left\{ x_k^{LM}, \vec{h}_{k,j}^{LM}, \vec{h'}_{k,j}^{LM}, \mid j = 1 \rightarrow L \right\} \quad (1)$$

$$h_{k,j}^{LM} = \left\{ \vec{h}_{k,j}^{LM}; \vec{h'}_{k,j}^{LM} \right\} \quad (2)$$

By disavowing the forward and backward biLM iteration directions using Equation 2 and substituting it in equation 1,

$$R_k = \left\{ h_{k,j}^{LM} \mid j = 0 \rightarrow N \right\} \quad (2)$$

Where $h_{k,0}^{LM}$ is the token layer for each biLSTM layer ELMo merged all layers to generate a word vector R as $ELMo_k = \{E(R_k; \Theta_e)\}$. The domain ∂ specific weight of ELMo is calculated as

$$ELMo_k^{\partial} = \delta^{\partial} \sum_{j=0}^{L} s_j^{\partial} h_{k,j}^{LM} \quad (3)$$

Where s^{∂} are the softmax normalized weights and δ^{∂} refers to the scalar parameter. The enhanced ELMo architecture is given in Figure II.

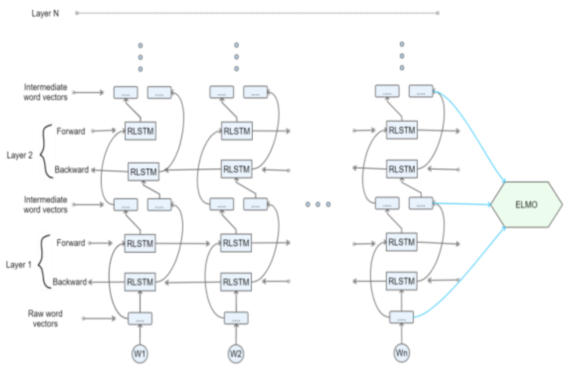

Figure II: Enhanced ELMo Architecture

By adding a crisis lexicon weight reformer logic to the standard LSTM model, EECWV module intercepts are made better. Figure III represents this modified LSTM as a Reformed LSTM (RLSTM). For every word vector at the disposal position by the existing forget gate, the sustainability of those vectors is increased by doubling the weight while one of the words exists in the crisis lexicon. Let ω be the set of weights of word vectors ω_i, represented as $\omega = \{\omega_0, \omega_1, \dots \omega_n\}$ with the condition $0 \le \omega_i \le 1$, where n is the maximum number of word vectors. The presence of the crisis-related word is ensured by the crisis lexicon reformer equation 4. The reformed LSTM Architecture is illustrated in Figure iii.

$$\forall i = 1 \to n$$
$$:= \begin{cases} 2\omega_i \; if(status(\omega_i) is\ disposal\ and\ word(w_i) \epsilon crisis\_lexican \\ \omega_i \; otherwise \end{cases} \quad (4)$$

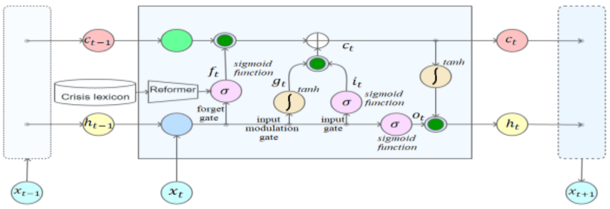

Figure III: Reformed LSTM

The suggested EECWV module is formulated by substituting the RLSTM for every conventional LSTM in the ELMo model. This weight retention for vectors containing crisis-related terms can endure for longer, resulting in an improved context-sensitive model. The equations for input, output, and forget gates are given in (Eq. 9, 10,11).

3.2 EnELMo CNN Architecture (ECA)

CNN is often used extensively for image categorization [33][34]. Since texts are single dimension character arrays sharing the characteristics of time series data, 1D-kernel is used for the convolution process. Generating vectors through word embeddings from language models enables the enhanced contextual feature extraction through CNN architecture. The steps of this module are explained using algorithm III – ECA algorithm. The architecture of this module is shown in figure IV.

Algorithm III: ECA(vector vm)
begin
Input the feature vector
vmwm<- 1D- Convolutiol_net(vm)
Kernel size(k) <- Equation(5)
Vmwmkm<- ReLU(vmwm, kernel km)
c<-MaxPool (vmwmkm)
return(cm)
end

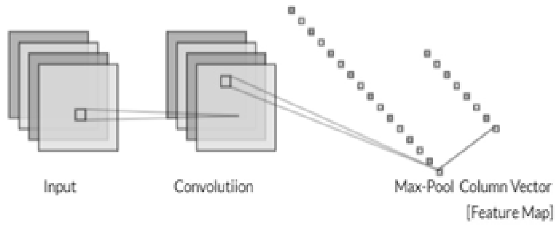

Figure IV: ECA Architecture

The word to vector conversion by enhanced ELMo ensures a more dense representation of context-sensitive input texts. There is also a greater probability of getting tangled in more than one embedding with similar contextual properties in the same direction. Therefore, a single dimension convolutional process takes place here. The procedure of 1D convolutional operation for extracting feature maps is represented in Figure V below.

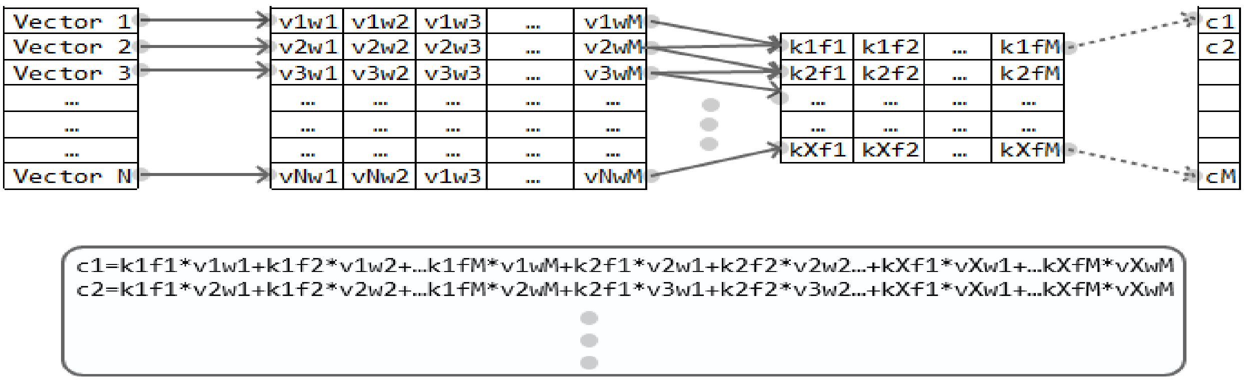

Figure V: Enhanced ELMo CNN Architecture

In Figure V, v1, v2, v3…vN refers to the input vectors supplied by the EECWV module, w1, w2…wM refers to the weights assigned to the vectors, k1, k2 … k X refers to the kernels, and f1, f2,…,f M refers the element values of the kernels. The result of the convolutions are represented as c1, c2,…cM. The convolution operation is the multiplication and addition cross input matrix and filter matrix resulting in a feature map(V_nW_n). All the input elements are covered using a sliding window of the filter matrix with a stride of 1[35]. The kernel size S_{kernel} influences the accuracy of the classification directly [36]. Therefore, in it is calculated dynamically in the proposed ECA model by Equation 11,

$$S_{kernel} = \max\left(sizeof(V_1), sizeof(V_2), \dots sizeof(V_N)\right) + sizeof(padding) \quad (5)$$

where $V_1, V_2, \dots V_N$ are the embedding vectors in which N refers to the number of vectors. The kernel size S_{kernel} comprehends the values of X and M. The output of this CNN layer is the column vector(c_{m}), which is the input text's feature map for further classification. The advantages of the above ELMo-CNN model are better detection of pertinent patterns from the text, such as bigrams, trigrams, and n-grams. The independent placement property of CNN makes it possible to detect these features from a sentence irrespective of the keywords' positions.

3.3 Crisis word vector-specific CNN and RNN stacks (CWV-CRNN)

CNN and RNN stacks are introduced here to acquire higher accuracy classification results of the streaming temporal text data. The output c1,c2,…cM of the ECA module is fed to the input section of the CWV-CRNN. The contextual convolutional layer (C-CN) is the most critical layer in the CRNN module. This module performs a number of iterations of the convolution over the sentence matrix. The C-CN provides a new interpretation of the meaning for the words by utilizing the crisis word lexicon connected to this module [30]. The context region for updating the current position is determined by the filter size m and the preset iteration step T : (m − 1)T + 1. After going through the non-linear activation function and adding the input, the C-CN attains non-linear word relationships, enhancing the meaning. An output unit at the ith index of the jth feature map at layer step t is defined by the equation: (6).

$$f(w) = \left(\Gamma \sum_{i=1}^{N}((W_i \otimes \omega) + b_i)\right) \quad (6)$$

Where $W_1, W_2, \dots W_N$ are vector weight matrices, $b_1, b_2, \dots b_N$ are corresponding biases and $\otimes$ is the elementwise multiplication operator. When the dot product of the embedded word vectors is used to detect commonalities between them, the interdependence of words for a specific class can be determined.
The CNN architecture used in this phase is given in Figure VI. The process of this module is listed below as algorithm VI – CWV-CRNN algorithm.

Algorithm IV - CWV-CRNN(feature_map)
Begin
//ensemble context aware convolutional net.
Conv_map <- convolutional_net(feature_map, crisis word lexicon)
Rectified_map<-relu(Conv_map)
conv_map<- convolutional_net(rectified_map)
Rectified_map<-relu(conv_map)
Reduced_map<-maxpool(rectified_map)
Context_map<-dense(reduced_map)
//classification determinator
P(c)<- Equation (6)
Classification_index <- Equation(7)
if Classification_index < 0.5
then
classification_index= rnn(context_map)
end if
//classification output
z<-label;

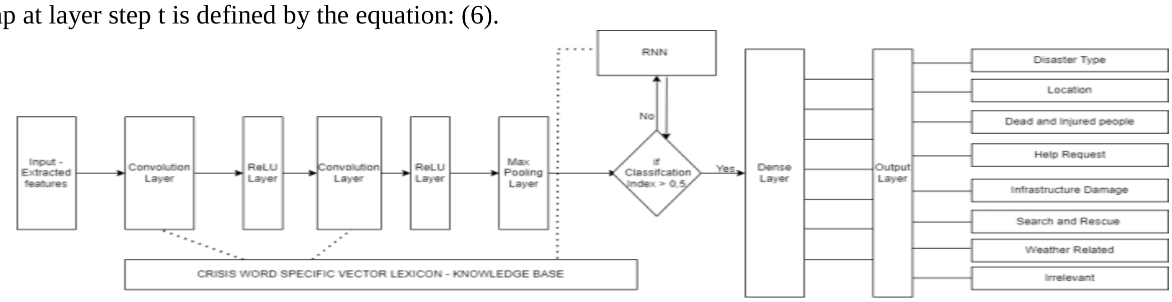

Figure VI: Architecture of CWV-CRNN Module

The sigmoid activation function is utilised at the dense layer of the model with the probability of a class (c_t) as Bernoulli distribution (Eq. 15 , 16). Now the probabilities of each class are independent of the other class probabilities. Based on the activation function, the threshold value is set to 0.5. At this stage, the output of this section(z) is highly accurate, and the CWV-CRNN module skips the subsequent RNN stack to conserve processing time. (Equation. 7) is used to decide whether the results are pertaining to the accuracy or not.

$$z = \begin{cases} highly\ accurate\ if \max(\Gamma_1, \Gamma_2 \dots \Gamma_8) \geq 2\sigma_\Gamma \\ negative\ otherwise \end{cases} \quad (7)$$

Where $\Gamma_1 \rightarrow \Gamma_8$ are the acquired classification index of desired classifications, σ_Γ indicates the standard deviation of every classification index value. If the output is less than the desired index, it is passed to the RNN module to improvise the classification index. RNN has obtained much attention because of its ability to preserve the sequence information over time. The embedding of the target tweet content and the contextual features are concatenated before sending to the LSTM layer. This generates compositions for entire sentences using RNN for the content and context data, which helps the sentences to a higher dimensional space and an output as numerical values for the context based on the content[37]. In this paper, RNN is utilized to capture past and future information. So, the output of the i-th word is shown in the following equation (8).

$$hi = \vec{h}_i \oplus \vec{\mathrm{h}}_i \quad (8)$$

Here, $\oplus$ denotes the concatenation of hidden states of the forward LSTM and the backward LSTM. The RNN stack equations for input, output, and forget gates of CWV-CRNN module using the equations (9), (10), (11), (12), (13), and (14).

$$i_t = \Gamma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (9)$$

$$f_t = \Gamma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \quad (10)$$

$$o = \Gamma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (11)$$

$$c_t = f_i c_{t-1} + i_t ReLU(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (12)$$

$$h_t = o_t ReLU(c_t) \quad (13)$$

$$y_t = w_{\vec{c}y}\vec{c}_t + w_{\vec{c}'y}\vec{c'}_t + b_y \quad (14)$$

The CWV-CRNN module receives N number of input vectors and produces 8 classification index values by which the classifications are determined either by positive or negative by comparing with the threshold value τ. Classification type notations and their equivalent classification labels are described in Table ii. The classification determinator equation is given below.

$$P\left(\frac{c_t}{x_i}\right) = \frac{1}{1+\exp(-z_t)} \quad (15)$$

$$\forall i = 1 \rightarrow 7 \coloneqq \Gamma_{r_i} = \begin{cases} Positive\ if\ \Gamma_i \geq \tau \\ Negative\ otherwise \end{cases} \quad (16)$$

Table II: Representations of Labels in CRNN Module

	Label
$\boldsymbol{\Gamma_1}$	Disaster type
$\boldsymbol{\Gamma_2}$	Location
$\boldsymbol{\Gamma_3}$	Dead and Injured People
$\boldsymbol{\Gamma_4}$	Help Request
$\boldsymbol{\Gamma_5}$	Infrastructure damage
$\boldsymbol{\Gamma_6}$	Search and Rescue
$\boldsymbol{\Gamma_7}$	Weather-related
$\boldsymbol{\Gamma_8}$	Non-relevant information

4 Experimental Setup

A computer powered by Intel Core i7, 3770K 3.5GHz Quad-Core processor with 16GB RAM and 1TB storage, is used to develop and test the proposed method. A dedicated User Interface (UI) is designed using Visual Studio Integrated Development Environment [37]. The UI is responsible for loading the dataset and evaluating the proposed method's performance. Visual C++ [38] programming language is used to code the functional modules of the proposed method. Existing and proposed methods are evaluated sequentially, and the performance metrics are logged into corresponding memory references. The UI is also designed to fetch the evaluation metric results from memory and export them as a report file. Graphs are generated in a separate window for all classifications. The UI can get the user's dynamic Twitter posts and classify them. CrisisNLP Twitter dataset [39] is used to measure the classification performance of the EWECNN method. Since the proposed method is context-sensitive, the preprocessing for input has only a negligible impact on the results. Therefore, the basic preprocessing of removing URLs, links, similes, and emoticons is done in this work to preserve processing complexity. Dataset is randomly divided into ten equal parts for the further process.

4.1 Dataset

We utilise data from the CrisisNLP standard dataset of tweets [35, 36]. The sources are tweets sent during several natural disasters and annotated by paid employees and volunteers. The dataset includes, among other occurrences, earthquakes, floods, and typhoons.. Tweets are categorized into many informative categories such as urgent needs, contribution offers, infrastructure damage, dead or injured persons, and irrelevant or unrelated class. A single-line description for each category is in Table iii.

Table III: Description of the classes in the datasets

Class	**Description**
Disaster_type	Disaster event
Location	Disaster location
Dead_and_Injured	Reports of deaths, and injuries, of the people during disaster
Help_Request	Messages containing donations (food, shelter, services, etc.) or volunteering offers
Infrastructure_damage	Reports of infrastructure and utility damage
Search_and_Rescue	Messages of search operations and rescue operations during crisis

Weather_related	Messages containing helpful information about the disaster event that does not fit in one of the above classes
irrelevant	Irrelevant or not informative, or not beneficial for crisis response

4.2 Classification

Dataset is split into 80% of the dataset as the training set and 20% as the test set. The validation set is made up of 20% of the training set. The word vectors were initialized using pre-trained crisis word embeddings using EMTerms. Multi-classification of Twitter texts is conducted using the Enhanced – Elmo – CNN feature extraction approach and ensemble stack of CRNN. The classification estimator of the ensemble stack improves the model's performance by enhancing the accuracy of label prediction and shortening the processing time. The entire dataset is divided into 10 equal parts, and the values for every 10 percent of the data are input into the model and recorded in the report.

5 Results and Discussion

There are 8 customarily used classifications as Disaster type, location, Dead and Injured people, Help request, Infrastructure damage, Search and Rescue, Weather-related and non-relevant information are targeted in this work. The Twitter NLP Crisis dataset is classified, and its classification performance parameters such as Accuracy, Precision, F-Score, and G-Mean [39] are discussed in this section.

5.1 Performance metrics of the proposed EWECNN Model with related models.

The metrics of the proposed model for multi-label classifications of the microblog text for the disaster events rescue are depicted with plots and discussed in detail in this section. Overall performance metrics scores of the proposed method and the existing methods discussed in section 2 are represented using table iv.

Table IV: Performance metrics score of the proposed EWECNN Model with the other methods.

Methods	Accuracy	Precision	F1-Score	G-Mean
MC_BDACSA	81.41	81.29	81.39	81.39
MC_PAMSA	79.76	79.79	79.76	79.76
MC_CRICCD	88.05	87.87	88.03	88.03
MC_MASC	83.00	83.13	83.02	83.02
MC_BMSC	85.58	85.28	85.54	85.54
MC_EWECNN	**90.46**	**90.43**	**90.46**	**90.46**

The above table – iv clearly shows the proposed model – EWECNN has higher accuracy than other methods discussed in this work. The proposed EWECNN method uses context-specific feature mapping and context-based classification using the EMTerms lexicon, which has improved the multi-label categorization of the tweets more than the other methods discussed here. In the other methods, the CRICCD method, which uses word embedding for feature vectorization, has better accuracy than the PAMSA, MASC, BMSC, and BDACSA methods discussed in section 2.

5.1.1 Accuracy

Accuracy refers to the classification results' proximity to the actual classified labels. The formula for calculating accuracy is (TP+TN)/(TP+TN+FP+FN) [40]. Figure VII shows a thorough Accuracy log comparison graph of the proposed model with existing approaches. Table iv shows the calculated accuracy values for classifications performed by existing methods and the proposed model. As per the observed results, the proposed EWECNN method achieved the highest accuracy average of 93.49%, which is 12.4% higher than the successive performer CRICCD. The ranking of the classification methods based on the accuracy are EWECNN, CRICCD, BMSC, MASC, BDACSA, and PAMSA with 93.5% 89.0 %, 86.6 %, 85.1 %, 83.0 %, and 81.10 % respectively.

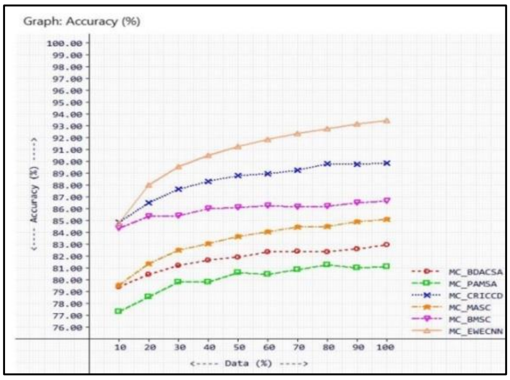

Figure VII Accuracy log comparison graph of the proposed EWECNN model with existing approaches

5.1.2 Precision

Precision is directly proportional to a classification algorithm's quality. A higher precision value refers higher quality of the algorithm. Precision is calculated using the formula TP/(TP+FP). A comparison of the proposed EWECNN's precision scores with the existing methods is represented in figure VIII. The score is listed in table iii. Based on the observation results, the performance rank sequence for proposed and existing methods in terms of precision is EWECNN, CRICCD, BMSC, MASC, BDACSA, and PAMSA, with values of 93.47 % ,89.55 %, 86.33 %, 83.76 %, 81.93 %, and

80.42 % respectively. The proposed EWECNN approach achieves the greatest precision of 93.47%, which is 3.9 percent greater than the closest precision score of CRICCD. During the trial, it was discovered that the proposed EWECNN approach outperformed the other methods regarding the classification-wise precision average.

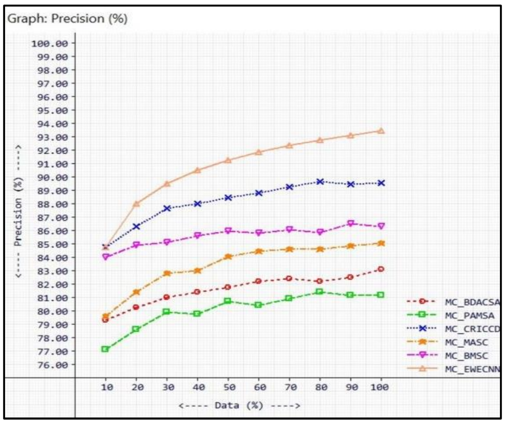

Figure VIII Comparison plot of the proposed EWECNN model with existing approaches for the performance metrics - Precision

5.1.3 F1-Score

The quality of a classification algorithm is directly proportional to the F-Score index. F-Score is calculated using the equation TP/(TP+1/2(FP+FN)). Table iii compares the F1-Score of the proposed model with the existing approaches.Classification wise F-Score averages of EWECNN, CRICCD, BMSC, MASC, BDASCA and PAMSA are 92.99%, 89.84%, 86.61%, 85.12%, 83.00% and 81.11% respectively. The proposed EWECNN approach achieves the greatest F1-Score of 93.49%, which is 3.15% greater than the closest precision score of CRICCD. Figure IX shows a comparative graph of the F-Score values of the proposed method with other methods discussed in this work.

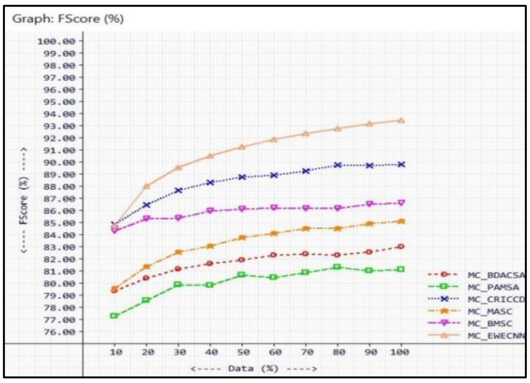

Figure IX: A Comparative graph of the F1-Score metric of the proposed EWECNN method with other methods

5.1.4 Geometric Mean(G-Mean)

The G-Mean is a statistic metric that evaluates the proportional performance of majority and minority classifications. Even if negative instances are correctly categorized, a low G-Mean suggests poor performance in classifying positive cases. This value is essential for preventing overfitting the negative class and underfitting the positive class. G-Mean is calculated using the formula √(TP/(TP+FP)×TP/(TP+FN)). Table iii depicts the G-Mean of the proposed method and existing methods. The highest GM value of 93.29% is achieved by the EWECNN method, which is 3.44% higher than the nearest achiever, CRICCD. Performance ranking sequence in terms of G-Mean is EWECNN, CRICCD, BMSC, MASC, BDACSA, and PAMSA, with the values 93.29%, 89.85%, 86.61%, 85.12%, 83.00%, and 81.11% given in order from the best score of G-Mean. The comparison graph is given below in Figure X.

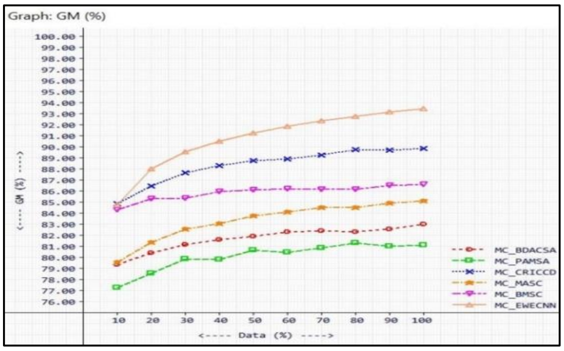

Figure X: Comparison of G-Mean metrics of the proposed EWECNN model with other models

Based on the observed result, it is understood that the proposed EWECNN method secures higher performance scores in terms of Accuracy, Precision, Sensitivity, Specificity, F1-Score, and G-Mean.

5.5 Performance analysis of the multi-label classification of the tweets using the proposed model.

The proposed EWECNN model multi-classifies the tweet text under eight categories. The suggested model's cumulative report analysis for all eight classes is discussed in this section. Table v shows the actual labels as well as the categorized labels

Table V: Count of Actual labels and Predicted labels

Class	#Actual labels	#Predicted Labels
Disaster type	6,418	5969
Location	5365	4882
Dead and Injured	2,437	2266
Help Request	3,683	3425
Infrastructure damage	3,288	3058
Search and Rescue	6,178	5746
Weather-related	10,059	9119

Irrelevant	12137	13300

Table vi shows the classification efficiency of the labels using the proposed method with the performance metrics – Accuracy, Precision, F-Score, and G-Mean. The tweets are assigned with these labels during a disaster event by the proposed EWECNN model. These labels aid in faster rescue operations.

Table VI: Performance metrics of classification of the labels by the proposed EWECNN Model.

S.No	Classification Labels	Accuracy	Precision	F1-Score	G-Mean
1	Disaster type	91.82	91.42	91.57	92.03
2	Location	92.29	91.78	92.26	92.26
3	Dead and Injured	93.39	93.37	93.39	93.39
4	Help Request	91.16	92.11	91.34	91.66
5	Infrastructure damage	93.94	93.12	92.95	92.95
6	Search and Rescue	93.08	92.96	93.07	93.07
7	Weather-related	94.16	93.53	94.16	94.14
8	Irrelevant	93.86	93.82	93.75	93.80

The above depicts the categorization of the tweet texts based on the labels. Among the eight labels, the categorization of help requests has the maximum accuracy than other labels. and has the least accuracy score among the labels. The overall accuracy average of the multi-classification of the label is 93.3% which is higher than the nearest model CRICCD. Figure XI presents the comparison graph on the performance metrics of the proposed EWECNN model.

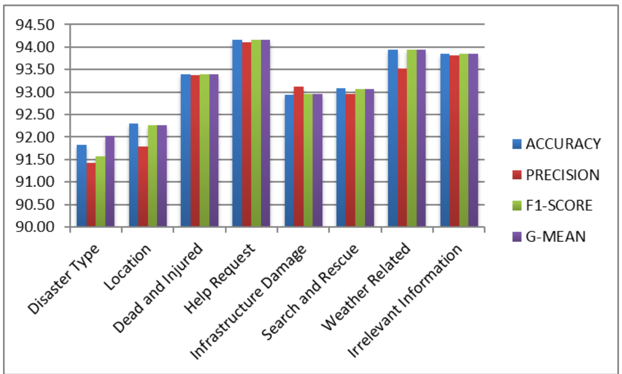

Figure XI: Performance analysis of the multi-label classification of the tweets using the proposed model

According to the plot presented above, the label requesting assistance has higher metrics than the other labels. When classifying the type of disaster, 91.82 percent accuracy was achieved. The proposed method achieves superior performance than the other method that has been discussed. These models employs baseline classifiers, perceptron classifiers, and deep learning classifiers for the classification of tweets .

This study demonstrates that using context-specific ELMo in conjunction with an ensemble of 1D-CNNs and RNNs is an effective way to improve multi-label classification and reduce processing time.

6 Conclusion

At present, the contribution of social media during any crisis is notably comparable with conventional media. At the same time, the ratio of irrelevant content in social media is much higher than in other conventional media. By incorporating a suitable classification procedure to deal with social media data, the impact of the irrelevant data can be reduced meaningfully. In this work, an improved classifier is introduced in the name 'EWECNN' to parse the Twitter data meticulously and classify them into given categories. The introduction of a crisis lexicon influenced LSTM into ELMo, Context-sensitive variable kernel CNN, and competent classification estimator based CRNN Stack are the novelties that contributed to this work. Though several existing methodologies are available to perform the multi-classification task, the novelties introduced in the proposed modules such as EECWV, ECA, and CRV-CRNN ensure higher classification accuracy. As per the experiments carried out with the benchmark dataset, it is asserted that the performance of the proposed EWECNN method has surpassed other methods. Due to this work's novelty and performance improvements, EWECNN can be used lively to support society during the crisis in a better way.

7 References

[1] [1] Olteanu A, Vieweg S and Castillo C. What to expect when the unexpected happens: Social media communications across crises., Proceedings of the 18th ACM conference on computer-supported cooperative work & social computing.ACM;.994–1009,2015.

[2] Gralla E, Goentzel J and Van de Walle B. Understanding the information needs of field-based decision-makers in humanitarian response to sudden-onset disasters., ISCRAM; 2015.

[3] Y. Agarwal, D. K. Sharma, and R. Katarya, Sentiment/Opinion Review Analysis: Detecting Spams from the good ones!, 2019, 4th International Conference on Information Systems and Computer Networks (ISCON),

557-563,2019. http://doi.org/10.1109/ISCON47742.2019.9036249.

[4] Imon Banerjee, Yuan Ling, Matthew C. Chen, Sadid A. Hasan, Curtis P. Langlotz, Nathaniel Moradzadeh, Brian Chapman, Timothy Amrhein, David Mong, Daniel L. Rubin, Oladimeji Farri, Matthew and P. Lungren, Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification, Artificial Intelligence in Medicine, 97, 79-88, 2019. https://doi.org/10.1016/j.artmed.2018.11.004

[5] Manzhu Yu, Qunying Huang, Han Qin, Chris Scheele and Chaowei Yang, Deep learning for real-time social media text classification for situation awareness – using Hurricanes Sandy, Harvey, and Irma as case studies, in International Journal of Digital Earth, 1230-1247, 2019. http://doi.org/10.1080/17538947.2019.1574316.

[6] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, A. Mehmood, and M. T. Sadiq, Document-Level Text Classification Using Single-Layer Multisize Filters Convolutional Neural Network, IEEE , 8, 42689 - 42707, 2020, http://doi.org/10.1109/ACCESS.2020.2976744.

[7] H. Ma, Y. Li, X. Ji, J. Han, and Z. Li, MsCoa: Multi-Step Co-Attention Model for Multi-Label Classification, IEEE Access, vol. 7, pp. 109635-109645, 2019, http://doi.org/10.1109/ACCESS.2019.2933042.

[8] Alsaedi, N., Burnap, P. and Rana, O. Can We Predict a Riot? Disruptive Event Detection Using Twitter, ACM Transactions on Internet Technology, 17(2), 18, 2017. https://doi.org/10.1145/2996183.

[9] Starbird, K., Palen, L., Hughes A.L., and Vieweg, S. Chatter on the red: what hazards threat reveals about the social life of microblogged information. Proceedings of CSCW. ACM, 241–250, 2010. https://doi.org/10.1145/1718918.1718965.

[10] Vieweg, S., Castillo, C. and Imran, M. Integrating social media communications into the rapid assessment of sudden onset disasters. In Social Informatics, Springer, 444–461, 2014. https://doi.org/10.1007/978-3-319-13734-6_32

[11] Verma, S., Vieweg, S., Corvey, W., Palen, L., Martin, J., Palmer, M., Schram, A., and Anderson, K. , Natural Language Processing to the Rescue? Extracting "Situational Awareness" Tweets During Mass Emergency. Proceedings of the International AAAI Conference on Web and Social Media, 5(1), 385-392., 2021. Retrieved from https://ojs.aaai.org/index.php/ICWSM/article/view/14119.

[12] Imran, M., Castillo, C., Lucas, J., Meier, P., and Vieweg, S., Aidr: Artificial intelligence for disaster response, in Proceedings of the 23rd International Conference on World Wide Web, ACM. pp. 159–162, 2014. https://doi.org/10.1145/2567948.2577034.

[13] Sreenivasulu and M., Sridevi, M., Mining informative words from the tweets for detecting the resources during disaster, International Conference on Mining Intelligence and Knowledge Exploration, Springer, 348–358, 2017. https://doi.org/10.1007/978-3-319-71928-3_33.

[14] Rudra, K., Ganguly, N., Goyal, P.,and Ghosh, S.,. Extracting and summarizing situational information from Twitter social media during disasters. ACM Transactions on the Web (TWEB) 12(17), 2018. https://doi.org/10.1145/3178541.

[15] Rudra, K., Ghosh, S., Ganguly, N., Goyal, P., and Ghosh, S., Extracting situational information from microblogs during disaster events: a classification-summarization approach, in Proceedings of the 24th ACM International Conference on Information and Knowledge Management, 583–599, 2015. https://doi.org/10.1145/2806416.2806485.

[16] Caragea, C., Silvescu, A. and Tapia, A.H., Identifying informative messages in disaster events using convolutional neural networks, in International Conference on Information Systems for Crisis Response and Management., 2016.

[17] Nguyen, T.D., Al-Mannai, K., Joty, S.R., Sajjad, H., Imran, M., and Mitra, P. , Rapid Classification of Crisis-Related Data on Social Networks using Convolutional Neural Networks. ArXiv, abs/1608.03902, 2016.

[18] Al-Rasheed, Amal A. "Finding Influential Users in Social Networking Using Sentiment Analysis." Informatica, 46(5), Mar. 2022. https://doi.org/10.31449/inf.v46i5.3829.

[19] Etaiwi, Wael & Suleiman, Dima and Awajan, Arafat,. Deep Learning Based Techniques for Sentiment Analysis: A Survey. Informatica. 45. 89-96. ,2021.

https://doi.org/10.31449/inf.v45i7.3674.

[20] Gaafar, Alaa Sahl, Jasim Mohammed Dahr, and Alaa Khalaf Hamoud.,Comparative Analysis of Performance of Deep Learning Classification Approach Based on LSTM-RNN for Textual and Image Datasets., Informatica 46(5), 2022. https://doi.org/10.31449/inf.v46i5.3872.

[21] Nguyen, D.T., Al Mannai, K.A., Joty, S., Sajjad, H., Imran, M.and Mitra, P., Robust classification of crisis-related data on social networks using convolutional neural networks, Eleventh nternational AAAI Conference on Web and Social Media., 2017.

[22] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. and Dean, J., Distributed representations of words and phrases and their compositionality, Advances in neural information processing systems, 3111–3119, 2013.

[23] Madichetty, S. and Sridevi, M., Detecting informative tweets during disaster using deep neural networks,, 11th International Conference on Communication Systems & Networks . OMSNETS), 709–713, 2019.

[24] Rodrigues, A.P.and Chiplunkar, N.N., A new big data approach for topic classification and sentiment analysis of Twitter data", Evol. Intel. 2019. https://doi.org/10.1007/s12065-019-00236-3.

[25] M. Bouazizi and T. Ohtsuki, A Pattern-Based Approach for Multi-Class Sentiment Analysis in Twitter, in IEEE Access, (5), 20617-20639, 2017. http://doi.org/10.1109/ACCESS.2017.2740982.

[26] Sreenivasulu, Madichetty and Sridevi, Improved Classification of Crisis-Related Data on Twitter using Contextual Representations, Procedia Computer Science, 167,962-968, 2020. https://doi.org/10.1016/j.procs.2020.03.395.

[27] Janjua, S. H., Siddiqui, G. F., Sindhu, M. A., and Rashid, U. Multi-level aspect based sentiment classification of Twitter data: using hybrid approach in deep learning. PeerJ. Computer science, 7, e433. , 2021. https://doi.org/10.7717/peerj-cs.433.

[28] Krawczyk B., McInnes B.T. and Cano A. Sentiment Classification from Multi-class Imbalanced Twitter Data Using Binarization. In: Martínez de Pisón F., Urraca R., Quintián H., Corchado E. (eds) Hybrid Artificial Intelligent Systems. HAIS 2017. Lecture Notes in Computer Science, vol 10334. Springer, Cham., 2017. https://doi.org/10.1007/978-3-319-59650-1_3

[29] Olteanu, C. Castillo, F. Diaz, and S. Vieweg, Crisislex: A lexicon for collecting and filtering microblogged communications in crises, Proceedings of the 8th International AAAI Conference on Weblogs and Social Media (ICWSM" 14), no. EPFL-CONF-203561, 2014.

[30] L.Mou and X. X. Zhu, Learning to Pay Attention on Spectral Domain: A Spectral Attention Module-Based Convolutional Network for Hyperspectral Image Classification, IEEE Transactions on Geoscience and Remote Sensing, 58(1), 110-122, 2020,. http://doi.org/10.1109/TGRS.2019.2933609.

[31] Y. Sun, B. Xue, M. Zhang, G. G. Yen, and J. Lv, Automatically Designing CNN Architectures Using the Genetic Algorithm for Image Classification, IEEE Transactions on Cybernetics, 50(9) . 3840-3854, 2020. http://doi.org/10.1109/TCYB.2020.2983860.

[32] Cui, Renhao & Agrawal and Gagan & Ramnath, Rajiv. Tweets can tell: activity recognition using hybrid gated recurrent neural networks., Social Network Analysis and Mining.,2020. https://doi.org/10.1007/s13278-020-0628-0.

[33] D. Hu and B. Krishnamachari, Fast and Accurate Streaming CNN Inference via Communication Compression on the Edge, IEEE/ACM Fifth International Conference on Internet-of-Things Design and Implementation (IoTDI), 157-163, 2020. http://doi.org/10.1109/IoTDI49375.2020.00023.

[34] Shin, Joongbo & Kim, Yanghoon & Yoon, Seunghyun & Jung and Kyomin. Contextual-CNN: A Novel Architecture Capturing Unified Meaning for Sentence Classification. 2018 IEEE International Conference on Big Data and Smart Computing,491-494, 018., 2018. https://doi.org/10.1109/bigcomp.2018.00079.

[35] Lan, Yangyang & Hao, Yazhou & Xia, Kui & Qian, Buyue and Li, Chen, Stacked Residual Recurrent Neural Networks With Cross-Layer Attention for Text Classification. IEEE Access. PP. 1-1.2020. https://doi.org/10.1109/access.2020.2987101.

[36] https://visualstudio.microsoft.com/

[37] https://www.udacity.com/blog/2020/02/microsoft-visual-c-review.html.

[38] https://crisisnlp.qcri.org/

[39] HaCohen-Kerner Y, Miller D, Yigal Y. The influence of preprocessing on text classification using a bag-of-words representation. PLOS ONE 5(5):e0232525,. 2020. DOI 10.1371/journal.pone.0232525.

[40] José-Ramón Cano, Pedro Antonio Gutiérrez, Bartosz Krawczyk, Michał Woźniak, Salvador García, "Monotonic classification: An overview on algorithms," in performance measures and data sets, Neurocomputing, 341, 168-182, 2019. https://doi.org/10.1016/j.neucom.2019.02.024.-182, ISSN 0925.